

Ser. No. 10/518,226
Internal Docket No. PU020288

IN THE CLAIMS:

1. (previously presented) A method of correcting errors in a data block having N rows and X columns, comprising:
 - performing a parity check for each one of said N rows of said data block;
 - performing a parity check for each one of said X columns of said data block, the parity check for each of said N rows and said X columns for detecting bit-level errors;
 - identifying at least one bad byte for said data block using a byte-level error detection process;
 - identifying, from said parity check for each one of said N rows of said data block, from said parity check for each one of said X columns of said data block and from said at least one identified bad bytes for said data block, at least one error in said data block; and
 - correcting each one of said at least one identified error in said data block.
2. (previously presented) The method of claim 1, wherein said at least one bad byte for said data block is identified using at least one of an 8B/10B encoding process and an 8B/10B decoding process.
3. (original) The method of claim 2, wherein N equals 32.
4. (previously presented) The method of claim 1, wherein each one of said at least one identified bad byte is located in one of said N rows of said data block.
5. (previously presented) The method of claim 4, wherein identifying at least one error in said data block further comprises:
 - identifying at least one suspect bit for each one of said at least one bad byte of said data block, each one of said at least one identified suspect bit being located in one of said X columns of said data block; and
 - determining, for each one of said at least one suspect bit, if a parity bit for a corresponding one of said X columns of said data block confirms that said suspect bit is suspect.

Ser. No. 10/518,226
Internal Docket No. PU020288

6. (previously presented) The method of claim 5, wherein correcting each one of said at least one identified error in said data block further comprises:

for each confirmed suspect bit, determining if at least one additional bit in said corresponding one of said X columns of said data block is also suspect; and

if no other bit in said corresponding one of said X columns of said data block is also suspect, correcting said confirmed suspect bit.

7. (previously presented) The method of claim 4, and further comprising identifying at least one suspect bit for each one of said at least one bad byte of said data block, each one of said at least one identified suspect bit being located in one of said X columns of said data block.

8. (previously presented) The method of claim 7, wherein identifying at least one suspect bit for each one of said at least one bad byte for said data block further comprises:

performing a parity check for each row of said data block in which one or more of said at least one bad byte is located; and

identifying suspect bits for each said parity checked row;

wherein said identified suspect bits for said parity checked row are said identified suspect bits for said bad byte.

9. (previously presented) The method of claim 8, wherein identifying at least one error in said data block further comprises:

identifying at least one suspect bit for each one of said at least one bad byte of said data block, each one of said at least one identified suspect bit being located in one of said X columns of said data block; and

determining, for each one of said at least one suspect bit, if a parity bit for a corresponding one of said X columns of said data block confirms that said suspect bit is suspect.

10. (previously presented) The method of claim 9, wherein correcting each one of said at least one identified error in said data block further comprises:

Ser. No. 10/518,226
Internal Docket No. PU020288

for each confirmed suspect bit, determining if at least one additional bit in said corresponding one of said X columns of said data block is also suspect; and
if no other bit in said corresponding one of said X columns of said data block is also suspect, correcting said confirmed suspect bit.

11. (previously presented) The method of claim 10, and further comprising:
for each said corrected suspect bit, correcting said suspect bit of said parity byte for a corresponding row of said data block.

12. (currently amended) A method of correcting errors in a data block, comprising:
(a) identifying at least one suspect bit in said data block, each one of said at least one suspect bit indicating a possible error in said data block;
(b) classifying each one of said at least one suspect bit as either a confirmed error bit or as an unconfirmed error bit; and
(c) correcting each one of said at least one suspect bit classified as a confirmed error bit,

wherein said identifying step identifies the at least one suspect bit using information derived from a byte-level error detection process and information derived from a bit-level error detection ~~process~~ process, and

wherein said confirmed error bits are identified using a combination of information derived from 8B/10B encoding of said data block, information derived from parity encoding along each row of said data block and information derived from parity encoding along each column of said data block.

13. (previously presented) The method of claim 12, and further comprising:
(d) repeating (a) through (c) until all errors in said data block have been corrected.

14. Cancelled.

15. (currently amended) The method of claim 14 12, wherein identifying at least one suspect bit of said data block further comprises identifying at least one bad byte of said data block using information derived from said 8B/10B encoding of said data block.

Ser. No. 10/518,226
Internal Docket No. PU020288

16. (previously presented) The method of claim 15, wherein classifying each one of said at least one suspect bit as either a confirmed error bit or as an unconfirmed error bit further comprises:

identifying, using information derived from parity encoding of a row of said data block in which a first one of said at least one bad byte is located, at least one suspect bit for said first one of said at least one bad byte.

17. (previously presented) The method of claim 16, wherein a first one of said at least one suspect bit for said first one of said at least one bad byte is classified as either a confirmed error bit or as an unconfirmed error bit using information derived from parity encoding of a column of said data block in said first suspect bit is located.

18. (previously presented) The method of claim 12, and further comprising the steps of:

applying a pre-selected number of iterations of an error-correction routine to said data block;

said error-correction routine being comprised of (a), (b) and (c).

19. (original) The method of claim 18, wherein said pre-selected number is two.

20. (previously presented) The method of claim 19, wherein said confirmed error bits are identified using a combination of information derived from 8B/10B encoding of said data block, information derived from parity encoding along each row of said data block and information derived from parity encoding along each column of said data block.

21. (previously presented) The method of claim 20, wherein identifying at least one suspect bit of said data block further comprises identifying at least one bad byte of said data block using information derived from said 8B/10B encoding of said data block.

22. (previously presented) The method of claim 21, wherein classifying each one of said at least one suspect bit as either a confirmed error bit or as an unconfirmed error bit

Ser. No. 10/518,226
Internal Docket No. PU020288

further comprises:

identifying, using information derived from parity encoding of a row of said data block in which a first one of said at least one bad byte is located, at least one suspect bit for said first one of said at least one bad byte.

23. (previously presented) The method of claim 22, wherein a first one of said at least one suspect bit for said first one of said at least one bad byte is classified as either a confirmed error bit or as an unconfirmed error bit using information derived from parity encoding of a column of said data block in said first suspect bit is located.